

## データベース第7回

第7章 正規化理論  
—更新時異状と情報無損失分解—

1

## 内容

- (次回第8章の)高次の正規化の準備
- 更新時異状
- 情報無損失分解
- 関数従属性と多値従属性
- 表をどうやって設計するか？ の参考になる考え方

2

## 第1正規形

- どの属性の値も集合であることはない
- 複数の属性にまたがる値はない
- これだけでは、よい表にはならないことがある

3

## 更新時異状 (update anomaly, 更新不整合)

注文

| 顧客名  | 商品名  | 数量 | 単価      | 金額        |
|------|------|----|---------|-----------|
| A商店  | テレビ  | 3  | 198,000 | 594,000   |
| Bマート | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 59,800  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800  | 29,800    |

- タプル挿入時異状
  - (-, 電子レンジ, -, 74,800, -)を挿入←キー制約から、無理！
- タプル削除時異状
  - (C社, 餅つき機, 1, 29,800, 29,800) ←重要なデータの喪失！
- タプル修正時異状
  - テレビの単価を198,000 から 148,000 に変更←修正大変！
  - C社からの注文を、餅つき機から洗濯機に変更←重要なデータの喪失！

4

## リレーションスキーマの分解

注文

| 顧客名  | 商品名  | 数量 | 単価      | 金額        |
|------|------|----|---------|-----------|
| A商店  | テレビ  | 3  | 198,000 | 594,000   |
| Bマート | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 59,800  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800  | 29,800    |

注文(=注文[顧客名, 商品名, 数量, 金額])

| 顧客名  | 商品名  | 数量 | 金額        |
|------|------|----|-----------|
| A商店  | テレビ  | 3  | 594,000   |
| Bマート | テレビ  | 10 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800    |

商品(=注文[商品名, 単価])

| 商品名  | 単価      |
|------|---------|
| テレビ  | 198,000 |
| 洗濯機  | 59,800  |
| 餅つき機 | 29,800  |

注文 = 注文[顧客名, 商品名, 数量, 金額] \* 注文[商品名, 単価]

5

## 更新時異状は解消される！

- タップル挿入時異状
  - (-, 電子レンジ, -, 74,800, -)を挿入 ← キー制約から, 無理！
- タップル削除時異状
  - (C社, 餅つき機, 1, 29,800, 29,800) ← 重要なデータの喪失！
- タップル修正時異状
  - テレビの単価を198,000 から 148,000 に変更 ← 修正大変！
  - C社からの注文を, 餅つき機から洗濯機に変更 ← 重要なデータの喪失！

注文(=注文[顧客名, 商品名, 数量, 金額])

| 顧客名  | 商品名  | 数量 | 金額        |
|------|------|----|-----------|
| A商店  | テレビ  | 3  | 594,000   |
| Bマート | テレビ  | 10 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800    |

商品(=注文[商品名, 単価])

| 商品名  | 単価      |
|------|---------|
| テレビ  | 198,000 |
| 洗濯機  | 59,800  |
| 餅つき機 | 29,800  |

注文 = 注文[顧客名, 商品名, 数量, 金額] \* 注文[商品名, 単価]

6

## リレーションスキーマの分解

- 分解して更新時異状を防ぐ
- one fact in one relation
- 分解前のリレーションが持っていた情報は失われてはならない
- リレーションを適当に(いい加減に)分解してしまうと, 自然結合をとっても元のリレーションが復元できないことに！

7

## リレーションスキーマの分解 悪い例

注文

| 顧客名  | 商品名  | 数量 | 単価      | 金額        |
|------|------|----|---------|-----------|
| A商店  | テレビ  | 3  | 198,000 | 594,000   |
| Bマート | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 59,800  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800  | 29,800    |

注文[顧客名, 商品名]

| 顧客名  | 商品名  |
|------|------|
| A商店  | テレビ  |
| Bマート | テレビ  |
| Bマート | 洗濯機  |
| C社   | 餅つき機 |

注文[商品名, 数量, 単価, 金額]

| 商品名  | 数量 | 単価      | 金額        |
|------|----|---------|-----------|
| テレビ  | 3  | 198,000 | 594,000   |
| テレビ  | 10 | 198,000 | 1,980,000 |
| 洗濯機  | 5  | 59,800  | 299,000   |
| 餅つき機 | 1  | 29,800  | 29,800    |

8

## リレーションスキーマの分解 悪い例

注文[顧客名, 商品名]

注文[商品名, 数量, 単価, 金額]

| 顧客名  | 商品名  |
|------|------|
| A商店  | テレビ  |
| Bマート | テレビ  |
| Bマート | 洗濯機  |
| C社   | 餅つき機 |

| 商品名  | 数量 | 単価      | 金額        |
|------|----|---------|-----------|
| テレビ  | 3  | 198,000 | 594,000   |
| テレビ  | 10 | 198,000 | 1,980,000 |
| 洗濯機  | 5  | 59,800  | 299,000   |
| 餅つき機 | 1  | 29,800  | 29,800    |

注文[顧客名, 商品名] \* 注文[商品名, 数量, 単価, 金額]

| 顧客名  | 商品名  | 数量 | 単価      | 金額        |
|------|------|----|---------|-----------|
| A商店  | テレビ  | 3  | 198,000 | 594,000   |
| A商店  | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | テレビ  | 3  | 198,000 | 594,000   |
| Bマート | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 59,800  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800  | 29,800    |

9

## リレーションスキーマの分解 悪い例

注文

| 顧客名  | 商品名  | 数量 | 単価      | 金額        |
|------|------|----|---------|-----------|
| A商店  | テレビ  | 3  | 198,000 | 594,000   |
| Bマート | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 59,800  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800  | 29,800    |

注文[顧客名, 商品名] \* 注文[商品名, 数量, 単価, 金額]

| 顧客名  | 商品名  | 数量 | 単価      | 金額        |
|------|------|----|---------|-----------|
| A商店  | テレビ  | 3  | 198,000 | 594,000   |
| A商店  | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | テレビ  | 3  | 198,000 | 594,000   |
| Bマート | テレビ  | 10 | 198,000 | 1,980,000 |
| Bマート | 洗濯機  | 5  | 59,800  | 299,000   |
| C社   | 餅つき機 | 1  | 29,800  | 29,800    |

10

## 情報無損失分解(定義)

リレーションスキーマ $R(X,Y,Z)$ , ここに $X, Y, Z$ は互いに素な属性集合とする, を2つの射影,  $R[X,Y]$ と $R[X,Z]$ に分解したとき,

$$R = R[X,Y] * R[X,Z]$$

が成立するならば, この分解は情報無損失(information lossless)であるという.

注:「互いに素」共通部分を持たない

$$X \cap Y = Y \cap Z = X \cap Z = \emptyset \text{ (空集合)}$$

注2: この性質は, リレーションスキーマに対して成立する.  
インスタンス(あるデータ)に対してのみではない

11

## 情報無損失分解(定理)

リレーションスキーマ $R(X,Y,Z)$ を2つの射影,  $R[X,Y]$ と $R[X,Z]$ に分解したとき,

$$R = R[X,Y] * R[X,Z]$$

が成立するための**必要十分条件**:

$R$ の任意のインスタンス $R$ に対して,  $t[X]=t'[X]$ を満たす $R$ の任意の2タプル $t$ と $t'$ につき, それらから構成される次の2タプル $w$ と $w'$ がまた $R$ のタプルであること. ここに,

$$w = (t[X,Y], t'[Z])$$

$$w' = (t'[X,Y], t[Z])$$

証明は教科書参照

12

## 多値従属性

リレーションスキーマ $R(X,Y,Z)$ に対して情報無損失分解(定理)の性質が成り立つとき,  $R$ に多値従属性(multi-valued dependency, MVD)  $X \twoheadrightarrow Y|Z$ が存在するという.

13

## 多値従属性の例

フライト

| フライト番号 | クルー名 | 乗客名 |
|--------|------|-----|
| 55     | P    | A   |
| 55     | S    | A   |
| 55     | P    | B   |
| 55     | S    | B   |
| 55     | P    | C   |
| 55     | S    | C   |
| 505    | P'   | A'  |
| 505    | S'   | A'  |

フライト

・パイロットや客室乗務員(クルー)  
・乗客

フライト55便

クルー: P, S

乗客: A, B, C

フライト505便

クルー: P', S'

乗客: A'

注: クルーと乗客は, たまたま  
同じフライトに乗り合わせただけ

フライト番号 $\twoheadrightarrow$ クルー名 | 乗客名

14

## 多値従属性の例

フライト

| フライト番号 | クルー名 | 乗客名 |
|--------|------|-----|
| 55     | P    | A   |
| 55     | S    | A   |
| 55     | P    | B   |
| 55     | S    | B   |
| 55     | P    | C   |
| 55     | S    | C   |
| 505    | P'   | A'  |
| 505    | S'   | A'  |

多値従属性

(フライト番号 $\twoheadrightarrow$ クルー名 | 乗客名)

$X \twoheadrightarrow Y|Z$

$t(x_1, y_1, z_1), u(x_1, y_2, z_2)$ という

タプルがあるなら, 必ず

$v(x_1, y_1, z_2), w(x_1, y_2, z_1)$ という

タプルがある, ということ.

フライト番号 $\twoheadrightarrow$ クルー名 | 乗客名

15

## 問題

- プロジェクトの表において, プロジェクト番号 $\twoheadrightarrow$ 社員番号 | ミーティング日の多値従属性があるとする. 表中に不足しているタプルを正確にすべて列挙しなさい.

プロジェクト

| プロジェクト番号 | 社員番号 | ミーティング日 |
|----------|------|---------|
| p1       | e2   | 木曜日     |
| p1       | e5   | 木曜日     |
| p2       | e1   | 月曜日     |
| p2       | e3   | 金曜日     |

## 解答

- プロジェクト番号→社員番号 | ミーティング日

プロジェクト

| プロジェクト番号 | 社員番号 | ミーティング日 |
|----------|------|---------|
| p1       | e2   | 木曜日     |
| p1       | e5   | 木曜日     |
| p2       | e1   | 月曜日     |
| p2       | e3   | 金曜日     |

(p2, e1, 金曜日)

(p2, e3, 月曜日)

## 情報無損失分解と多値従属性

リレーションスキーマ $R(X,Y,Z)$ がその二つの射影 $R[X,Y]$ と $R[X,Z]$ に情報無損失分解されるための必要かつ十分条件は $R$ に多値従属性 $X \twoheadrightarrow Y|Z$ が存在すること。

18

## 関数従属性

- 多値従属性の特殊な場合
- リレーションの情報無損失分解をするときに重要となるもの
- 第2正規形, 第3正規形, ボイス-コード正規形を規定するのにとても重要となるもの

19

## 関数従属性

リレーションスキーマ $R(X,Y,Z)$ に関数従属性 (functional dependency)  $X \rightarrow Y$ が存在するとは次の条件が成立するときをいう。

$R$ を $R$ の任意のインスタンスとすると,

$$(\forall t, t' \in R)(t[X]=t'[X] \Rightarrow t[Y]=t'[Y])$$

20

## 関数従属性

リレーションスキーマ $R(X, Y, Z)$ に関数従属性 (functional dependency)  $X \rightarrow Y$ が存在するとは次の条件が成立するときをいう.

$R$ を $R$ の任意のインスタンスとすると、  
 $(\forall t, t' \in R)(t[X]=t'[X] \Rightarrow t[Y]=t'[Y])$

- リレーション $R$ において、2つのタプルの属性 $X$ の値が同じであれば、属性 $Y$ の値も同じになる
  - 属性 $X$ が決まれば、属性 $Y$ も決まる

21

## 関数従属性の例

| 履修     |           |    |    |     |      |      |
|--------|-----------|----|----|-----|------|------|
| 学籍番号   | 科目        | 得点 | 評価 | 判定  | 担当教員 | 入学年  |
| 200100 | データベース    | 70 | A  | 合   | 福井   | 2020 |
| 200100 | プログラミング言語 | 80 | B  | 合   | 宮崎   | 2020 |
| 210123 | データベース    | 80 | A  | 合   | 福井   | 2021 |
| 210123 | 計算機システム   | 90 | A  | 合   | 石川   | 2021 |
| 210124 | データベース    | 20 | D  | 不合格 | 福井   | 2021 |
| 210124 | 計算機システム   | 50 | C  | 合   | 石川   | 2021 |
| 210124 | プログラミング言語 | 70 | B  | 合   | 宮崎   | 2021 |

- $f_1$ : {学籍番号, 科目} → 得点
- $f_2$ : {科目, 得点} → 評価 (得点だけでは評価は決まらない、と仮定)
- $f_3$ : 評価 → 判定
- $f_4$ : 学籍番号 → 入学年
- $f_5$ : 科目 → 担当教員

22

## 候補キー

リレーションスキーマ $R(A_1, A_2, \dots, A_n)$ の属性集合 $K$ が候補キーであるとは次の性質を満たすときをいう.

$R$ を $R$ の任意のインスタンスとして、

- $(\forall t, t' \in R)(t[K]=t'[K] \Rightarrow t=t')$ ,
- $K$ のどのような真部分集合 $H$ に対しても1.の性質は成立しない. ( $K$ は極小組)

1は、関数従属性 $K \rightarrow \{A_1, A_2, \dots, A_n\}$

候補キーを含む集合をスーパーキー(super key)という.  
 スーパーキーでは1.しか成り立たない.

23

## 候補キーと主キー(以前のスライドより)

| 社員   |      |    |     |       |
|------|------|----|-----|-------|
| 社員番号 | 社員名  | 給与 | 所属  | 健保番号  |
| 0650 | 山田太郎 | 50 | K55 | 80596 |
| 1508 | 鈴木花子 | 40 | K41 | 81403 |
| 0231 | 田中桃子 | 60 | K41 | 80201 |
| 2034 | 佐藤一郎 | 40 | K55 | 81998 |

キーはどれ？

- 社員番号
- 健保番号

キーとなる属性の組が複数ある場合  
 それらを**候補キー**という

その中の一つを**主キー**という  
 どれを主キーにするかに決まりはない

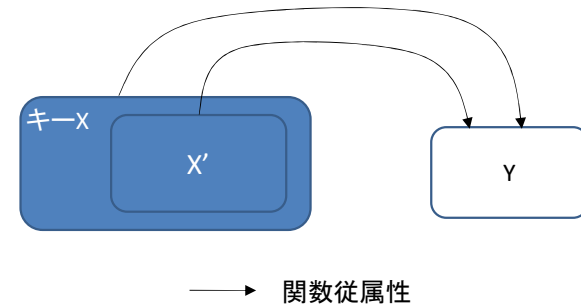
24

## 完全関数従属性

関数従属性  $X \rightarrow Y$  で、 $X$  の任意の真部分集合  $X'$  ( $X' \subset X$ ) について  $X' \rightarrow Y$  は成立しないとき、 $Y$  は  $X$  に完全関数従属 (fully functionally dependent) しているという。

25

## 部分的関数従属性



26

## 関数従属性と多値従属性

リレーションスキーマ  $R(X, Y, Z)$  に関数従属性 (FD)  $X \rightarrow Y$  が存在すれば、多値従属性 (MVD)  $X \twoheadrightarrow Y | Z$  が存在する。

27

## 証明

関数従属性  $X \rightarrow Y$  より、 $t[X] = t'[X]$  ならば  $t[Y] = t'[Y]$  なので、 $t[X, Y] = t'[X, Y]$ 。従って  $w = (t[X, Y], t'[Z]) = (t'[X, Y], t'[Z]) = t'$  となる。同様に  $w' = t$  となり、 $w$  も  $w'$  も  $R$  のタプルである。

$t[X] = t'[X]$  なるタプル  $t, t'$  について、 $w = (t[X, Y], t'[Z])$  と  $w' = (t'[X, Y], t[Z])$  が両方とも  $R$  のタプルということがわかる。

28

## 情報無損失分解の十分条件

リレーションスキーマ $R(X,Y,Z)$ がその2つの射影 $R[X,Y]$ と $R[X,Z]$ に情報無損失分解されるための十分条件は $R$ に関数従属性 $X \rightarrow Y$ が存在すること。

– 注: 必要条件ではない

29

## 関数従属性

- リレーションスキーマを定義するとき、関数従属性に注目することが重要
- スキーマ定義の時に関数従属性を把握しきれているか？ は難しい
- 関数従属性を把握するにはどうすれば？

30

## アームストロングの公理系

1.  $X$ を属性集合,  $Y$ を $X$ の部分集合とするなら  $X \rightarrow Y$ である. (反射律)
2.  $X \rightarrow Y$ かつ,  $Z$ を任意の属性集合とすると,  $X \cup Z \rightarrow Y \cup Z$ である. (添加律)
3.  $X \rightarrow Y$ かつ  $Y \rightarrow Z$ なら  $X \rightarrow Z$ である. (推移律)

31

## 関数従属性の推移律

リレーションスキーマ $R$ に関数従属性  $X \rightarrow Y$  と  $Y \rightarrow Z$  が存在したとする. このとき,  $X \rightarrow Z$  が成立する.

32



## 関数従属性の推移律の証明

いま,  $X \rightarrow Y$ かつ $Y \rightarrow Z$ なのに $X \rightarrow Z$ が成立しないとする. すると,  $R$ のあるインスタンス $R$ が存在して,  $R$ に少なくとも2タプル $t$ と $t'$ が存在して,  $t[X]=t'[X]$ なのに $t[Z] \neq t'[Z]$ となる. しかし,  $X \rightarrow Y$ なので $t[X]=t'[X]$ ならば $t[Y]=t'[Y]$ であり, さらに $Y \rightarrow Z$ なので $t[Y]=t'[Y]$ ならば $t[Z]=t'[Z]$ である. しかし, これは仮定に矛盾する. よって $X \rightarrow Z$ が成立する.

33

## アームストロングの公理系

- 関数従属性の集合 $F$ が与えられたら, アームストロングの公理系により導出されるものは確かにリレーションスキーマ $R$ 上での関数従属性である(健全性)
- $F$ が与えられたら, 公理系により, リレーションスキーマ $R$ 上で成立すべき関数従属性は全て導出できる(完全性)

34

## 関数従属性の例

履修

| 学籍番号   | 科目        | 得点 | 評価 | 判定  | 担当教員 | 入学年  |
|--------|-----------|----|----|-----|------|------|
| 200100 | データベース    | 70 | A  | 合   | 福井   | 2020 |
| 200100 | プログラミング言語 | 80 | B  | 合   | 宮崎   | 2020 |
| 210123 | データベース    | 80 | A  | 合   | 福井   | 2021 |
| 210123 | 計算機システム   | 90 | A  | 合   | 石川   | 2021 |
| 210124 | データベース    | 20 | D  | 不合格 | 福井   | 2021 |
| 210124 | 計算機システム   | 50 | C  | 合   | 石川   | 2021 |
| 210124 | プログラミング言語 | 70 | B  | 合   | 宮崎   | 2021 |

$f_1: \{\text{学籍番号, 科目}\} \rightarrow \text{得点}$   
 $f_2: \{\text{科目, 得点}\} \rightarrow \text{評価}$  (得点だけでは評価は決まらない, と仮定)  
 $f_3: \text{評価} \rightarrow \text{判定}$   
 $f_4: \text{学籍番号} \rightarrow \text{入学年}$   
 $f_5: \text{科目} \rightarrow \text{担当教員}$

35

## 関数従属性の例

$f_1: \{\text{学籍番号, 科目}\} \rightarrow \text{得点}$   
 $f_2: \{\text{科目, 得点}\} \rightarrow \text{評価}$  (得点だけでは評価は決まらない, と仮定)  
 $f_3: \text{評価} \rightarrow \text{判定}$   
 $f_4: \text{学籍番号} \rightarrow \text{入学年}$   
 $f_5: \text{科目} \rightarrow \text{担当教員}$

- $\{\text{学籍番号, 科目}\} \rightarrow \text{得点}$  (所与)
- $\{\text{学籍番号, 科目}\} \rightarrow \{\text{科目, 得点}\}$  (1. と添加律)
- $\{\text{科目, 得点}\} \rightarrow \text{評価}$  (所与)
- $\{\text{学籍番号, 科目}\} \rightarrow \text{評価}$  (2. と 3. と推移律)
- $\text{評価} \rightarrow \text{判定}$  (所与)
- $\{\text{学籍番号, 科目}\} \rightarrow \text{判定}$  (4. と 5. と推移律)

36

## $X^+$ を求めるアルゴリズム

- (ステップ1)  $X^{(0)} = X$  とおく.
- (ステップ2)  $X^{(i)} = X^{(i-1)} \cup \{A \mid A \in Z \wedge Y \rightarrow Z \in F \wedge Y \subseteq X^{(i-1)}\}$   
( $i \geq 1$ )
- (ステップ3) もし  $X^{(i)} = X^{(i-1)}$  なら  $X^+ = X^{(i-1)}$  とおく. そうでなければステップ2に行く.

37

## 候補キーを1つ見つけるアルゴリズム

- (ステップ1)  $K = \{A_1, A_2, \dots, A_n\}$  とおく.
- (ステップ2) 属性  $A_i \in K$  を選び,  $\{K - A_i\}^+$  を計算する. もし,  $\{K - A_i\}^+ = \{A_1, A_2, \dots, A_n\}$  ならば,  $K = K - A_i$  においてステップ2に戻る. そうでなければ,  $K$  が求める候補キーである.

38